UTILITY PATENT APPLICATION TRANSMITTAL

(Only for new nonprovisional applications under 37 CFR 1.53(b))

Attorney Docket No. 826.1636/JDH

First Named Inventor or Application Identifier:

Koujirou OGURA

Express Mail Label No.

APPLICATION ELEMENTS

See MPEP chapter 600 concerning utility patent application contents.

ADDRESS TO: **Assistant Commissioner for Patents**
Box Patent Application
Washington, DC 20231

1. [X] Fee Transmittal Form
2. [X] Specification, Claims & Abstract ..... [Total Pages: 25]
3. [X] Drawing(s) *(35 USC 113)* ..... [Total Sheets: 11]
4. [X] Oath or Declaration ..... [Total Pages: 3]
   a. [X] Newly executed (original or copy)
   b. [ ] Copy from a prior application (37 CFR 1.63(d)) *(for continuation/divisional with Box 17 completed)*
      i. [ ] DELETION OF INVENTOR(S)
      Signed statement attached deleting inventor(s) named in the prior application, see 37 CFR 1.63(d)(2) and 1.33(b).
5. [ ] Incorporation by Reference (usable if Box 4b is checked)
   The entire disclosure of the prior application, from which a copy of the oath or declaration is supplied under Box 4b, is considered as being part of the disclosure of the accompanying application and is hereby incorporated by reference therein.
6. [ ] Microfiche Computer Program *(Appendix)*
7. [ ] Nucleotide and/or Amino Acid Sequence Submission *(if applicable, all necessary)*
   a. [ ] Computer Readable Copy
   b. [ ] Paper Copy (identical to computer copy)
   c. [ ] Statement verifying identity of above copies

ACCOMPANYING APPLICATION PARTS

8. [X] Assignment Papers (cover sheet & document(s))
9. [ ] 37 CFR 3.73(b) Statement *(when there is an assignee)* [ ] Power of Attorney
10. [ ] English Translation Document *(if applicable)*
11. [X] Information Disclosure Statement (IDS)/PTO-1449[X] Copies of IDS Citations
12. [X] Preliminary Amendment
13. [X] Return Receipt Postcard (MPEP 503) *(Should be specifically itemized)*
14. [ ] Small Entity Statement(s) [ ] Statement filed in prior application, status still proper and desired.
15. [X] Certified Copy of Priority Document(s) *(if foreign priority is claimed) (Japanese Appln. 2000-094801, filed March 30, 2000.*
16. [ ] Other:

17. If a CONTINUING APPLICATION, *check appropriate box and supply the requisite information:*

[ ] Continuation [ ] Divisional [ ] Continuation-in-part (CIP) of prior application No: ___/_____

18. CORRESPONDENCE ADDRESS

21171
PATENT TRADEMARK OFFICE

Staas & Halsey

S&H Form (10/00)

NEW APPLICATION FEE TRANSMITTAL		Attorney Docket No.	826.1636/JDH
		Application Number	To Be Assigned
		Filing Date	November 16, 2000
AMOUNT ENCLOSED	**$910.00**	First Named Inventor	Koujirou OGURA

FEE CALCULATION (fees effective 10/01/00)

CLAIMS	(1) FOR	(2) NUMBER FILED	(3) NUMBER EXTRA	(4) RATE	(5) CALCULATIONS
	TOTAL CLAIMS	15 - 20 =	0	X $ 18.00 =	$ 0.00
	INDEPENDENT CLAIMS	5 - 3 =	2	X $ 80.00 =	160.00
	MULTIPLE DEPENDENT CLAIMS (any number; if applicable)			+ $270.00 =	0.00
			BASIC FILING FEE		710.00
			Total of above Calculations =		$ 870.00
	Surcharge for late filing fee, Statement or Power of Attorney ($130.00)				0.00
	Reduction by 50% for filing by small entity (37 CFR 1.9, 1.27 & 1.28).				- 0.00
			TOTAL FILING FEE =		$ 870.00
	Surcharge for filing non-English language application ($130.00; 37 CFR 1.52(d))				0.00
	Recordation of Assignment ($40.00; 37 CFR 1.21(h)(1))				40.00
			TOTAL FEES DUE =		$ 910.00

METHOD OF PAYMENT

[X] Check enclosed as payment.

[ ] Charge "TOTAL FEES DUE" to the Deposit Account No., below.

[ ] No payment is enclosed and no charges to the Deposit Account are authorized at this time.

GENERAL AUTHORIZATION

[X] If the above-noted "AMOUNT ENCLOSED" is not correct, the Commissioner is hereby authorized to credit any overpayment or charge any additional fees necessary to:

Deposit Account No.	19-3935
Deposit Account Name	STAAS & HALSEY LLP

[X] The Commissioner is also authorized to credit any overpayments or charge any additional fees required under 37 CFR 1.16 (filing fees) or 37 CFR 1.17 (processing fees) during the prosecution of this application, including any related application(s) claiming benefit hereof pursuant to 35 USC § 120 (e.g., continuations/divisionals/CIPs under 37 CFR 1.53(b) and/or continuations/divisionals/CPAs under 37 CFR 1.53(d)) to maintain pendency hereof or of any such related application.

SUBMITTED BY: STAAS & HALSEY LLP

Typed Name	James D. Halsey, Jr.	Reg. No.	22,729
Signature		Date	November 16, 2000

Docket No. 826.1636/JDH

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:	)	
	)	
Koujirou OGURA	)	
	)	Group Art Unit: To Be Assigned
Serial No.: To Be Assigned	)	
	)	
Filed: November 16, 2000	)	Examiner: To Be Assigned
	)	
For: SYSTEM AND METHOD FOR AUTOMATICALLY SETTING APPLET LATEST VERSION	)	

PRELIMINARY AMENDMENT

Assistant Commissioner of Patents and Trademarks
Washington, D.C. 20231

Sir:

Before examination of the above-identified application, please amend the application as follows:

IN THE CLAIMS:

Claim 11, line 1, change "claim 1 or 2" to --claim 1--.

Claim 12, line 1, change "claim 3 or y" to --claim 3--.

REMARKS

This Preliminary Amendment is submitted to improve the form of the claims as originally filed.

It is respectfully requested that this Preliminary Amendment be entered in the above-references application.

If any fees are required in connection with the filing of this Preliminary Amendment, please charge same to our Deposit Account No. 19-3935.

Respectfully submitted,

STAAS & HALSEY LLP

By: _____

James D. Halsey, Jr.
Registration No.: 22,729

700 Eleventh Street, N.W.
Washington, D.C. 20001
(202) 434-1500

Date: November 16, 2000

APPLICATION FOR

UNITED STATES LETTERS PATENT

SPECIFICATION

INVENTOR(s): Koujirou Ogura

Title of the Invention: SYSTEM AND METHOD FOR AUTOMATICALLY SETTING APPLET LATEST VERSION

SYSTEM AND METHOD FOR AUTOMATICALLY SETTING APPLET LATEST VERSION

Background of the Invention

Field of the Invention

The present invention relates to an applet latest version automatically setting system and method for automatically setting the latest version of an applet such as a Java applet, etc. at a request of a client.

Description of the Prior Art

An applet program operated on different pieces of software is used in the current computer technology. Especially, a Java applet is expressed in a Java language, incorporated into an HTML file, and executed on a Web server.

Conventionally, the above mentioned Java applet is downloaded by a server when a client invokes a Web page, and is cashed and used on the Java virtual machine of the Web browser. Normally, the Java applet is downloaded and used on the main memory of a client, but can be temporarily stored on a disk (hard disk, etc.) of a client, and then transmitted to the main memory for execution.

However, in the conventional system, after transmitting an applet to a client, the applet on the server side can be rewritten as the latest version. In this case, normally, the update of an applet is not notified to the client. Therefore, the client performs a process using an applet of an old version.

In such a case, the already cached and operated applet is updated by the browser using a button, and an applet can be read again. However, the server cannot predict when the version of an applet is updated, thereby performing an inefficient operation. In addition, depending on the setting of a browser, only an applet written to, for example, a disk can be read although a cached applet is re-read without updating the version of the applet.

Summary of the Invention

The present invention has been developed to solve the above mentioned problems, and aims at providing a system and a method for automatically setting the latest version of an applet by storing a program for updating the version in the method for the applet, and realizing the update of the

version of the applet in the process of the applet itself.

That is, the above mentioned purpose can be attained by the present invention by providing an applet latest version automatically setting system including: a server having an applet storage unit for storing an applet of the latest version, and a version number storage unit for storing the version number of an applet stored in the applet storage unit; and a client having a version number information storage unit for performing a process based on the applet read from the server, and storing the version number information about the applet, a comparison unit for reading the version number information about the applet of the latest version from the version number storage unit when a predetermined event occurs, and comparing the read information with the version number information stored in the version number information storage unit, and an update unit for reading the applet of the latest version from the applet storage unit and updating the applet when the comparison process outputs a non-matching result.

The applet stored in the applet storage unit is, for example, a program written in the Java

language, and is sequentially updated, thereby storing the applet of the latest version. In addition, the version number information storage unit stores the version number information about the applet of the latest version.

For example, the server storing the above mentioned applet of the latest version can be a Web server, and communications are established between the server and the Web browser set for the client.

Brief Description of the Drawings

FIG. 1 shows the configuration of the system of the applet latest version automatically setting system according to an embodiment of the present invention;

FIG. 2A shows the configuration of the memory area in which the version number information about the latest version is entered;

FIG. 2B shows an example of actual data written to a memory area;

FIG. 3 shows the configuration of the system according to an embodiment of the present invention which is applied to a sales office server and a window terminal;

FIG. 4 is a flowchart of the processing

operations according to an embodiment of the present invention;

FIG. 5 is a flowchart of the processing operations in the initial method;

FIG. 6 is a flowchart of the processing operations in the starting method;

FIG. 7 is a flowchart of the version number checking process;

FIG. 8 is a flowchart of the version number inquiring method;

FIG. 9 shows the link mechanism on the right;

FIG. 10 is an oblique view of the link mechanism; and

FIG. 11 shows an example of a case in which a storage medium is used.

Description of the Preferred Embodiment

The embodiments of the present invention are described below by referring to the attached drawings.

FIG. 1 shows the configuration of the applet latest version automatically setting system according to the an embodiment of the present invention. In FIG. 1, the system includes a server 1 and a client 2 connected to the server 1. A

plurality of clients 2 are practically connected to the server 1, but the connection to one client is shown in FIG. 1. A Web server 3 is set in the server 1. Memory 4 in which the applet program of the latest version is entered, and a version number file 5 in which the version number information about a corresponding latest applet is entered are set in the Web server 3. The applet entered in the memory 4 is a program for driving the Java, and the applet drives the Java virtual machine described later and performs a process.

The version number information about the latest version is stored in the version number file 5 in, for example, the format shown in FIG. 2. FIG. 2A shows the memory format of the version number file 5 configured by 4 bytes with the lower 2 bytes indicating version information and the higher 2 bytes indicating level information. For example, the version number 'V20L10' shown in FIG. 2B indicates the applet of the latest version having the version number is 20, and the level is 10. Therefore, in this case, as shown in FIG. 2B, the version area of the version number file 5 stores 14, and the level area stores 0A. The above mentioned level indicates, for example, a development number

in the same version. According to the present invention, it is not always necessary to include a level in the latest version.

On the client 2 side, the software of a Web browser 6, a Java virtual machine (VM) 7, and an applet 8 are set. The applet 8 is entered in a cache area 9. The version number information about the applet being operated is stored in a process unit 10. For example, the version number information storage area is set at a predetermined address.

The Web browser 6 displays the image of the data in the HTML (hyper text markup language) format provided from the Web server 3, and the applet (Java applet) 8 performs a process using the Java virtual machine 7. The above mentioned applet reads the applet of the latest version from the server 1 in the method described later, and updates the applet written in the cache area 9. A display (DSP) 2' displays according to the operation of the above mentioned applet.

FIG. 3 shows an example in which the relationship between the above mentioned server 1 and the client 2 is applied to the sales office server and the window terminal of a bank. A sales

office server 11 corresponds to the server 1, and a window terminal 12 corresponds to the client 2. The sales office server 11 has the Web server 3, the memory 4, and the version number file 5 as described above, and CORBR (common object request broker architecture) communications are established between the server and the window terminal 12. To attain this, a cooperative application 13 for establishing the CORBR communications is also installed.

In addition, the above mentioned Web browser 6 is mounted on the window terminal 12 so that the system can be activated by an operator 14. In addition, the applet used in this case relates to banking operations, and has the displays and functions required by the operator at the window of a bank.

In the applet latest version automatically setting system with the above mentioned configuration, the processing operations are performed as described below.

FIG. 4 is a flowchart of the processing operations according to the present invention, and shows the main process of an applet. In the following explanation, the Web server 3 of the

sales office server 11 which is the server 1, and the Web browser 6 of the window terminal 12 which is the client 2 have already been activated

First, an initializing process is performed (step (hereinafter expressed by S) 1). This initializing process (init) activates the Web browser 6, reads the applet from the server 1, and loads the applet of the latest version to the cache area 9 in the client 2 (window terminal 12). Then, practically, the initial method shown in FIG. 5 is performed. That is, the constant/variable initializing process is performed (S2), and the version number of the applet loaded on the cache area 9 is set. In this case, for example, when the current version of the applet is 'V20L10' as shown in FIG. 2, the data of '140A' is written. Then the applet initializing process is performed (S3). Next, the version number checking process is performed (S4). The version number checking process is performed when an applet is started, and is performed in the process of the applet starting method. FIG. 6 shows the process, and the version number checking process is performed in the starting method (S4'). Therefore, this process is described by referring to the version number

checking process (start).

FIG. 7 is a flowchart practically showing the version number checking process. First, the version number is inquired of the server 1 (sales office server 11) (S5). Then, it is compared with the version number information stored in the memory of the server 1 (S6). If the version number of the applet read to the client 2 is smaller than the version number of the server 1 (version number of client < version number of server), then the terminating process described later is performed (YES in S7).

FIG. 8 is a flowchart of the server version number inquiring process and the server version number reading process. This process is performed according to the server version number inquiring method. This process is performed by initializing the CORBA communications on the client 2 (window terminal 12) side (S5-1), and the client is connected to the server 1 (sales office server 11 (S5-2). Then, the version number information about the applet is read from the version number file 5 in the sales office server 11 (S5-3).

In this case, the version number file 5 in the server 1 (sales office server 11) is opened (S5-4),

and the version number file is read (S5-5). Then, the read version number information is written (S5-6), and the version number file is closed (S5-7). Then, the version number of the server is set in the out-parameter (S5-8), the server 1 (sales office server 11) is disconnected (S5-9), and the post-CORBA communications process is performed (S5-10).

Thus, the version number inquiring process is performed as described above. However, the version number checking process (start) is performed when the starting method process is performed after the applet initialization, and the version number of the applet read by the Web server 3 matches the version number of the server 1 (version number of client = version number of server). Therefore, the determination (S7) is N (NO). As a result, in the version number checking process in the starting method, an event waiting state is entered (S8).

Then, a normal transaction is started. For example, the client 2 (window terminal 12) requests the server 1 (sales office server 11) to perform a process online. Depending on the contents of the operation, the process is performed offline, and can be performed only by the window terminal 12.

During the process, the occurrence of an event is awaited (N in S8 and S9). When an event occurs (Y (YES) in S9), it is determined whether or not it is a user event (S10). If it is a user event, it is furthermore determined whether or not an online process is required (S11). A user event refers to a process requiring the above mentioned online process, and the client is connected to the server 1 (sales office server 11) to use the application of the server 1.

In this case, the version number checking process (S12) is first performed. The version number checking process (S12) is performed according to the flowchart shown in FIG. 7. That is, the version number is inquired of the server 1 (S5), the version number data is compared (S6), and the version number about the applet is compared with the version number of the server 1 to know whether of not the version number of the client is smaller than the version number of the server 1 (version number of client < version number of server) (S7). The version number inquiring process is performed according to the above mentioned version number inquiring method (S5-1 through S5-10 shown in FIG. 8).

As a result of the above mentioned process, the version number information read to the version number file is compared with the version number of the currently read applet. If the version numbers match each other, the user-specified event process is performed (S13), control is returned to the event wait state (S9). In this case, the version number of the applet entered in the server 1 (sales office server 11) is not updated, the version number of the server is equal to the version number of the applet of the client 2 (window terminal 12), the applet used in the current client 2 is the latest applet, the applet is not updated, and the current applet performs a terminal operations.

FIG. 9 shows a type of the above mentioned process. In this case, the version number memory of the server 1 (sales office server 11) stores the applet of the latest version, and the version number of the applet is not updated. Therefore, the version number information maintains as is. As a result, the above mentioned version number information inquiring process (1 shown in FIG. 9) is performed, it is determined whether or not the version numbers match each other, and the transaction starting process is performed (2 shown

in FIG. 9).

On the other hand, if the version number of the client is smaller than the version number of the server (Y (YES) in S7) in the above mentioned determining process (S7), then it is determined that the version number of the applet is updated, and the terminating process is performed (S14). This process is a destroying process (S14), the program of the applet stored in the cache area 9 is removed, and the applet of the latest version is read from the server 1 (S15).

The version number information about a corresponding version is read from the version number file 5, and written to the memory of the client 2. For example, updated data of '1500 or '140B' is written to the memory.

In the above mentioned process, when an event occurs, but not in an online process (N (NO) in S11), for example, when a process is performed with control passed to another Web page not using an applet, a corresponding user event process is performed (S16).

On the other hand, if the Web server 3 displays another Web page, or if the browser is displayed as an icon, an applet is in an inactive

state. If control is returned to the Web pace or a browser displayed as an icon is restored, then a system event occurs (N (NO) in S10). In this case, it is determined whether or not the above mentioned starting event is being performed (S17).

If the starting method is activated (YES in S17), then the above mentioned version number checking process (start) is performed. This process is also performed according to the flowchart shown in FIG. 7, the version number inquiring process (S5, S5-1 through S5-10) and the comparing process are performed (S6 and S7). If the version number of the client is smaller than the version number of the server, then it is determined that the version number of the applet has been updated, and the terminating process is performed (Y (YES) in S7, S14). Therefore, in this case, the applet stored in the cache area 9 is deleted, the applet of the latest version is read from the server 1 and stored (S15). In addition, the version number information about the corresponding version is read from the version number file 5, and written to the memory of the client 2.

FIG. 10 shows a type of the above mentioned process. The applet of the server 1 (sales office

server 11) is updated, and the version number information is different. Therefore, the version number information inquiring process (1 shown in FIG. 10) is performed, and the version number information is rewritten (2 shown in FIG. 10).

By performing the above mentioned process, the memory of the client 2 (window terminal 12) stores an applet of the latest version, and based on the applet of the latest version, an image and animation can be displayed.

When the above mentioned system event which is not a starting event occurs, it is determined whether or not it is a destroy event (S19). If it is a destroy event, the terminating process is performed (S20). The terminating process deletes the program of the applet stored in the main memory as described above, reads the applet of the latest version from the server 1, and stores it.

If it is not determined to be a destroy event (S19), a specified event process is performed (S21).

If an applet is set in an active state as described above according to the present invention, or if the version number of an applet is checked under a predetermined condition, for example, when a user event requiring the online process, and the

applet is not the latest version, then the applet is necessarily updated into the latest version. Therefore, the Web browser 6 can be operated by the applet of the latest version.

According to the present embodiment, the version number checking process is performed in the above mentioned starting method, etc., but is not limited to the described method.

The version number information according to the present embodiment contains the version information and the level information as shown in FIG. 2, but can be formed by the version information only.

According to the present embodiment, the server 1 of the system is the sales office server 11 of a bank, and the client 2 is the window terminal 12 of a bank. However, the present invention is not limited to the above mentioned bank system, but can be applied to other fields such as insurance and policy systems.

Furthermore, the applet latest version automatically setting system according to the present invention can also be realized using the system shown in FIG. 11. That is, as shown in FIG. 11, a CPU (computer) 15 performs a process on the

above mentioned system using a program (data) provided by a hard disk 16.

As shown in FIG. 11, not only the program (data) provided by the internal RAM and hard disk 16, but also data is transmitted to and received from an externally connected storage medium. For example, a process can be performed according to the program provided from a storage medium 17 such as a magnetic disk, a magnetic tape, a floppy disk, an optical disk, etc.

Furthermore, as shown in FIG. 11, a program (data)can be received from a provider 18 through a communications line.

As described above in detail according to the present invention, the Web browser can be operated always using an applet of the latest version.

In addition, the transaction reliability can be greatly improved when a client and a server cooperates with each other in a bank online system, etc. and an important transactions requiring high reliability is realized using a portable program such as a Java applet, etc.

What is claimed is:

1. An applet latest version automatically setting system, comprising:

a server having an applet storage unit storing an applet of a latest version, and a version number storage unit storing a version number of an applet stored in the applet storage unit; and

a client having a version number information storage unit performing a process based on the applet read from said server, and storing version number information about the applet, a comparison unit reading the version number information about the applet of the latest version from the version number storage unit when a predetermined event occurs, and comparing the read information with the version number information stored in the version number information storage unit, and an update unit reading the applet of the latest version from the applet storage unit and updating the applet when a comparison process outputs a non-matching result.

2. A server, comprising:

an applet storage unit updating an applet, and storing an applet of a latest version; and

a version number storage unit storing a version number of an applet stored in said applet storage unit, wherein

version number information is read from said version number storage unit at a version number information read request, and is transmitted to a client.

3. A client, comprising:

a version number information storage unit performing a process based on the applet read from said server, and storing version number information about the applet;

a comparison unit reading the version number information about the applet of the latest version from said version number storage unit when a predetermined event occurs, and comparing the read information with the version number information stored in said version number information storage unit; and

an update unit reading the applet of the latest version from said applet storage unit and updating the applet when a comparison process outputs a non-matching result.

4. The system according to claim 1, wherein said applet is a Java applet.

5. The server according to claim 2, wherein said applet is a Java applet.

6. The client according to claim 2, wherein said applet is a Java applet.

7. The system according to claim 4, wherein said predetermined event occurs when the applet becomes active.

8. The client according to claim 6, wherein said predetermined event occurs when the applet becomes active.

9. The system according to claim 1, wherein CORBA communications are established between said server and said client.

10. The system according to claim 1, wherein said server is a banking operation center, and said client is a branch office terminal device.

11. The system according to claim 1 or 2, wherein said version number information storage unit is main memory or a disk.

12. The client according to claim 3 or y, wherein said version number information storage unit is main memory or a disk.

13. An applet latest version automatically setting method, comprising the processes of:

storing an applet of a latest version in an applet storage unit in a server;

storing version number information corresponding to a version number of the applet in a version number storage unit in a server;

storing the version number information read from said version number storage unit in a version number information storage unit in a client;

reading a version number of an applet of a latest version from the version number storage unit when a predetermined event occurs, and comparing the read version number with the version number information stored in the version number information storage unit; and

reading the applet of the latest version from

the applet storage unit when a comparing process outputs a non-matching result, and updating the applet.

14. The system according to claim 13, wherein said applet is a Java applet.

15. A computer-readable storage medium storing a program used to direct a computer to perform the functions of:

storing an applet of a latest version in an applet storage unit in a server;

storing version number information corresponding to a version number of the applet in a version number storage unit in a server;

storing the version number information read from said version number storage unit in a version number information storage unit in a client;

reading a version number of an applet of a latest version from the version number storage unit when a predetermined event occurs, and comparing the read version number with the version number information stored in the version number information storage unit; and

reading the applet of the latest version from

the applet storage unit when a comparing process outputs a non-matching result, and updating the applet.

Abstract of the Disclosure

An applet latest version automatically setting system automatically sets the latest version of an applet such as a Java applet, etc. at a request of a client. Specifically, it stores in a method of an applet a program to update a version, realizes the update of the version of an applet in a process of the applet itself, and can necessarily use an applet of the latest version.

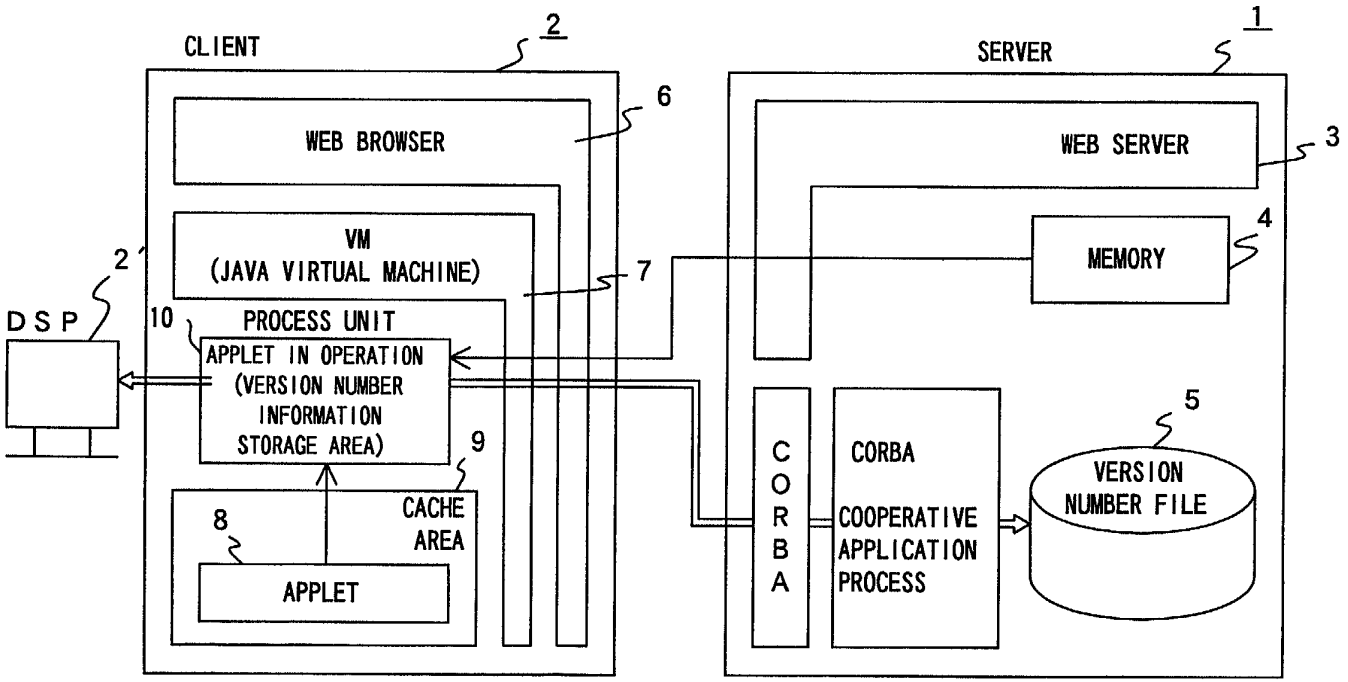


FIG. 1

FIG. 2A

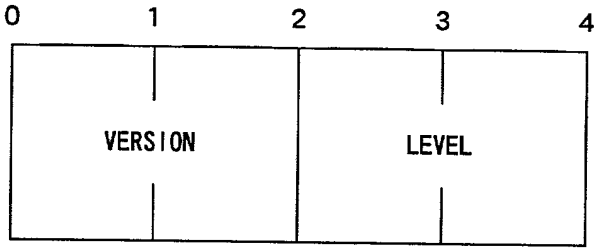
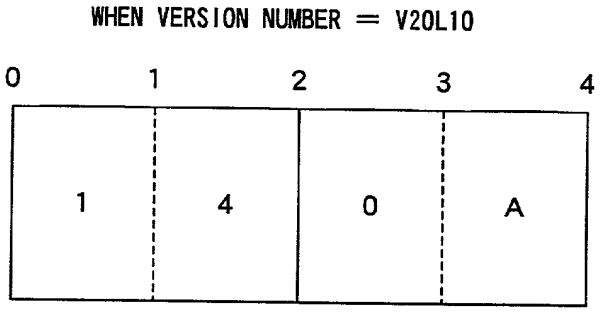


FIG. 2B



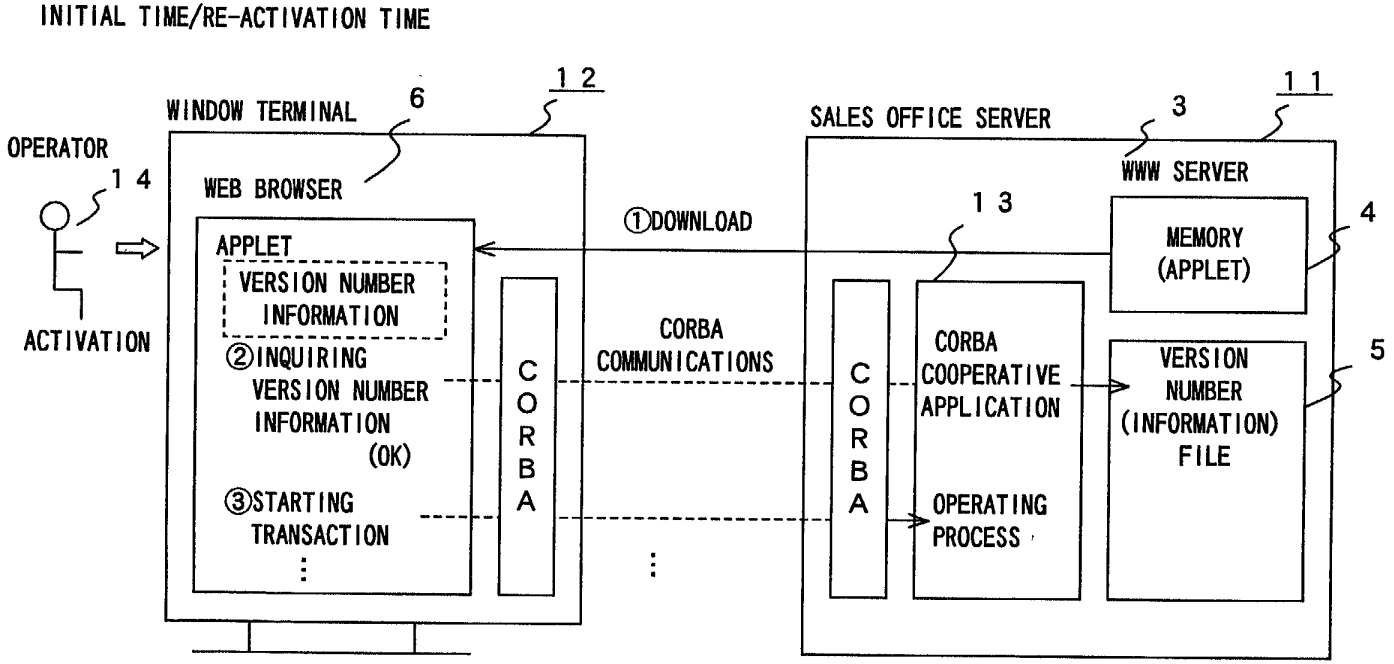


FIG. 3

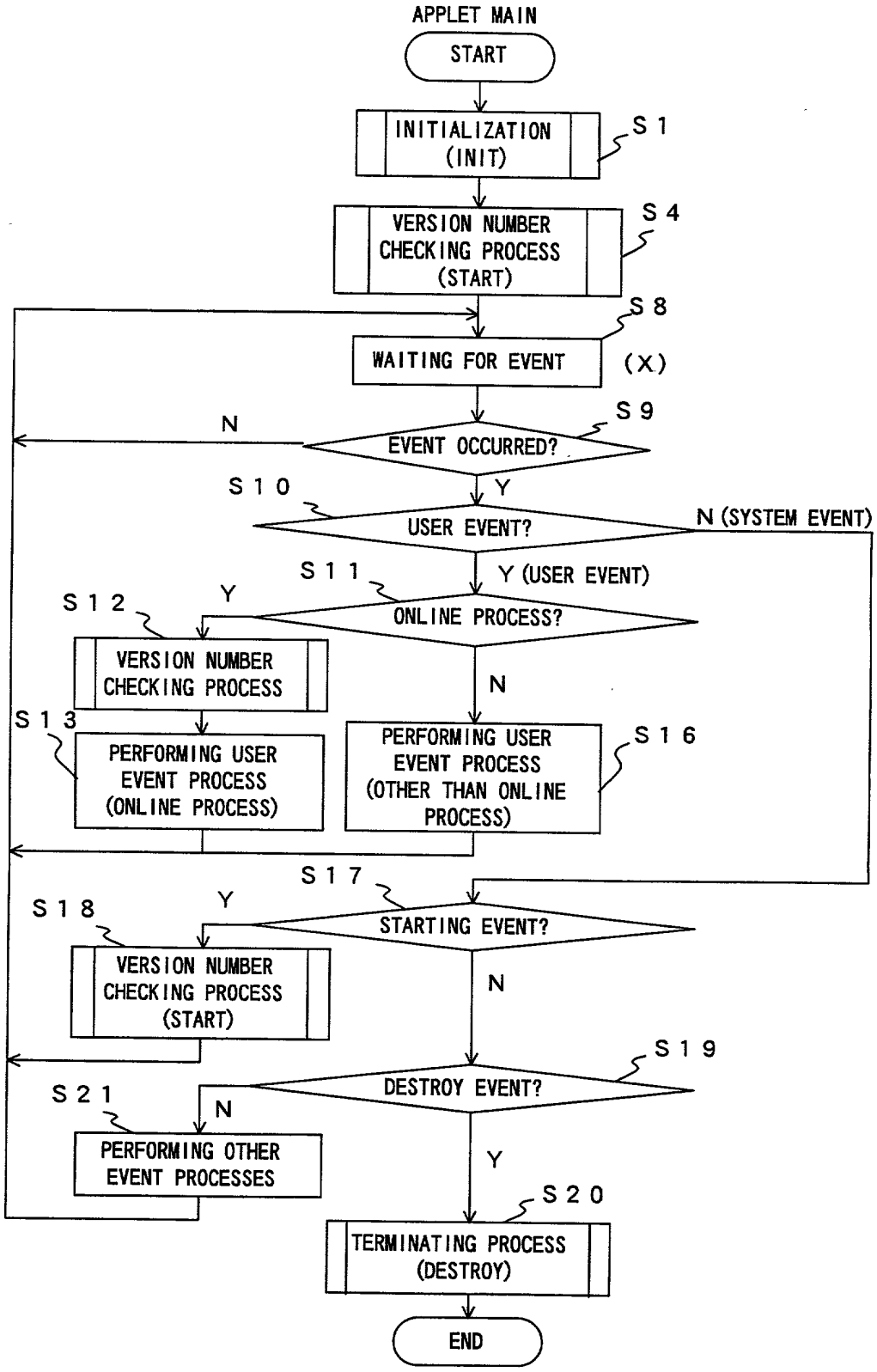

APPLET MAIN
START
INITIALIZATION (INIT)
S 1
VERSION NUMBER CHECKING PROCESS (START)
S 4
S 8
WAITING FOR EVENT
(X)
S 9
N
EVENT OCCURRED?
Y
S 10
USER EVENT?
N (SYSTEM EVENT)
S 11
Y (USER EVENT)
S 12
Y
ONLINE PROCESS?
VERSION NUMBER CHECKING PROCESS
N
S 13
PERFORMING USER EVENT PROCESS (ONLINE PROCESS)
PERFORMING USER EVENT PROCESS (OTHER THAN ONLINE PROCESS)
S 16
S 17
S 18
Y
STARTING EVENT?
VERSION NUMBER CHECKING PROCESS (START)
N
S 19
DESTROY EVENT?
S 21
N
PERFORMING OTHER EVENT PROCESSES
Y
S 20
TERMINATING PROCESS (DESTROY)
END

FIG. 4

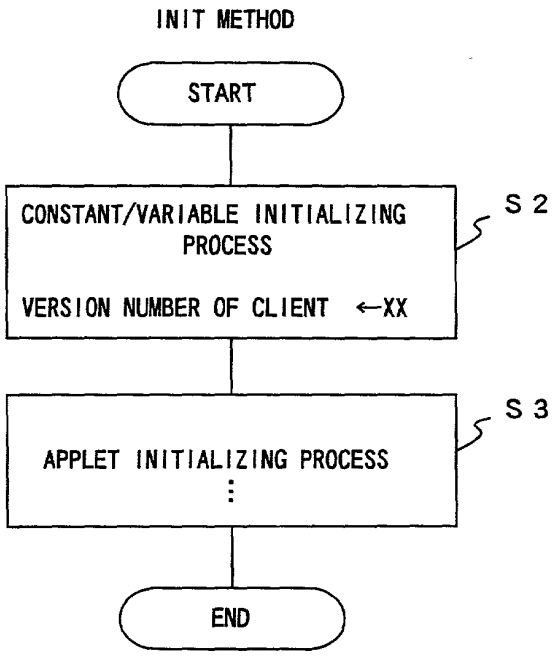

INIT METHOD
START
CONSTANT/VARIABLE INITIALIZING PROCESS
VERSION NUMBER OF CLIENT ←XX
S 2
APPLET INITIALIZING PROCESS
S 3
END

FIG. 5

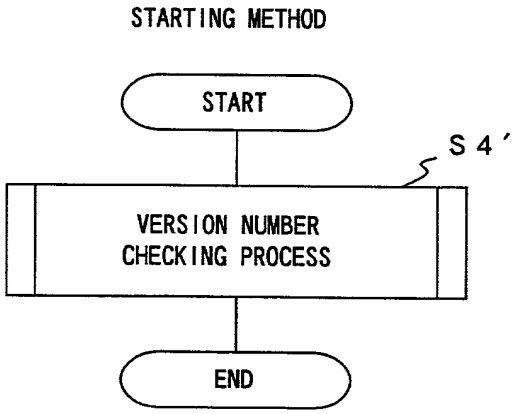
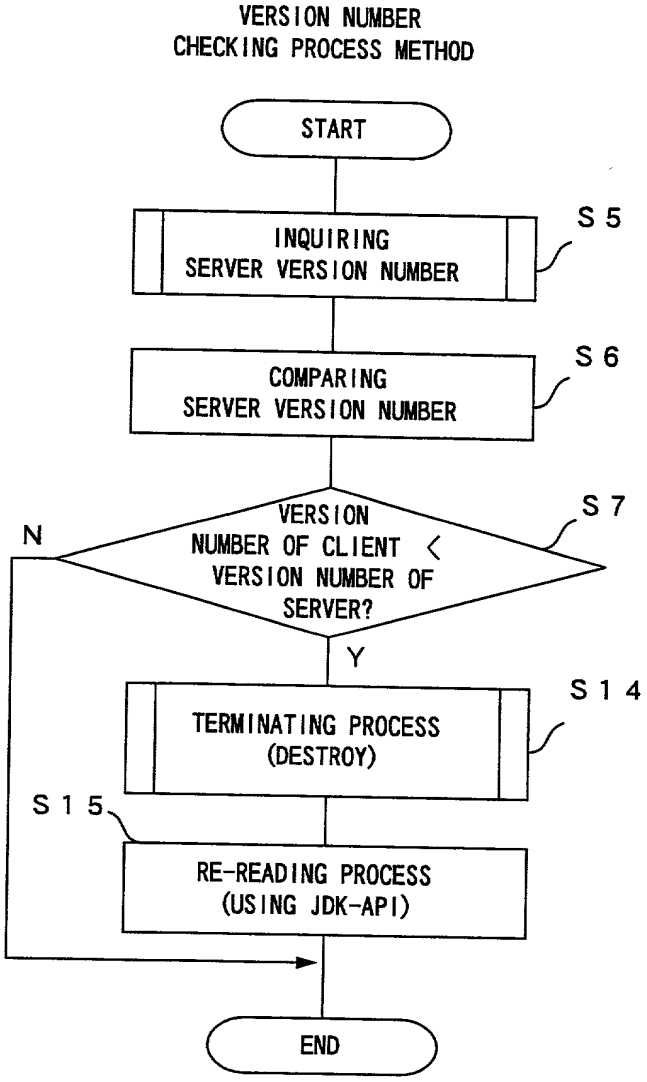

STARTING METHOD
START
S4'
VERSION NUMBER
CHECKING PROCESS
END

FIG. 6


VERSION NUMBER
CHECKING PROCESS METHOD
START
INQUIRING
SERVER VERSION NUMBER
S 5
COMPARING
SERVER VERSION NUMBER
S 6
VERSION
NUMBER OF CLIENT <
VERSION NUMBER OF
SERVER?
S 7
N
Y
TERMINATING PROCESS
(DESTROY)
S 1 4
S 1 5
RE-READING PROCESS
(USING JDK-API)
END

F I G. 7

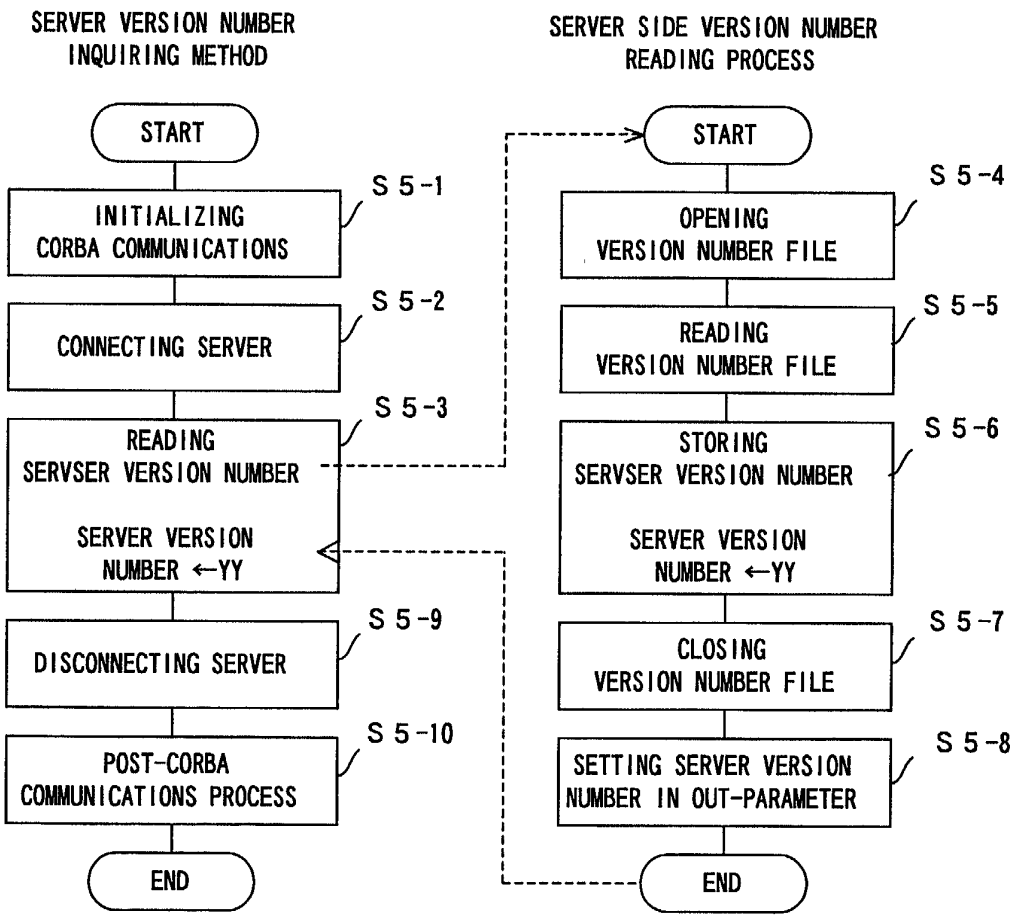


FIG. 8

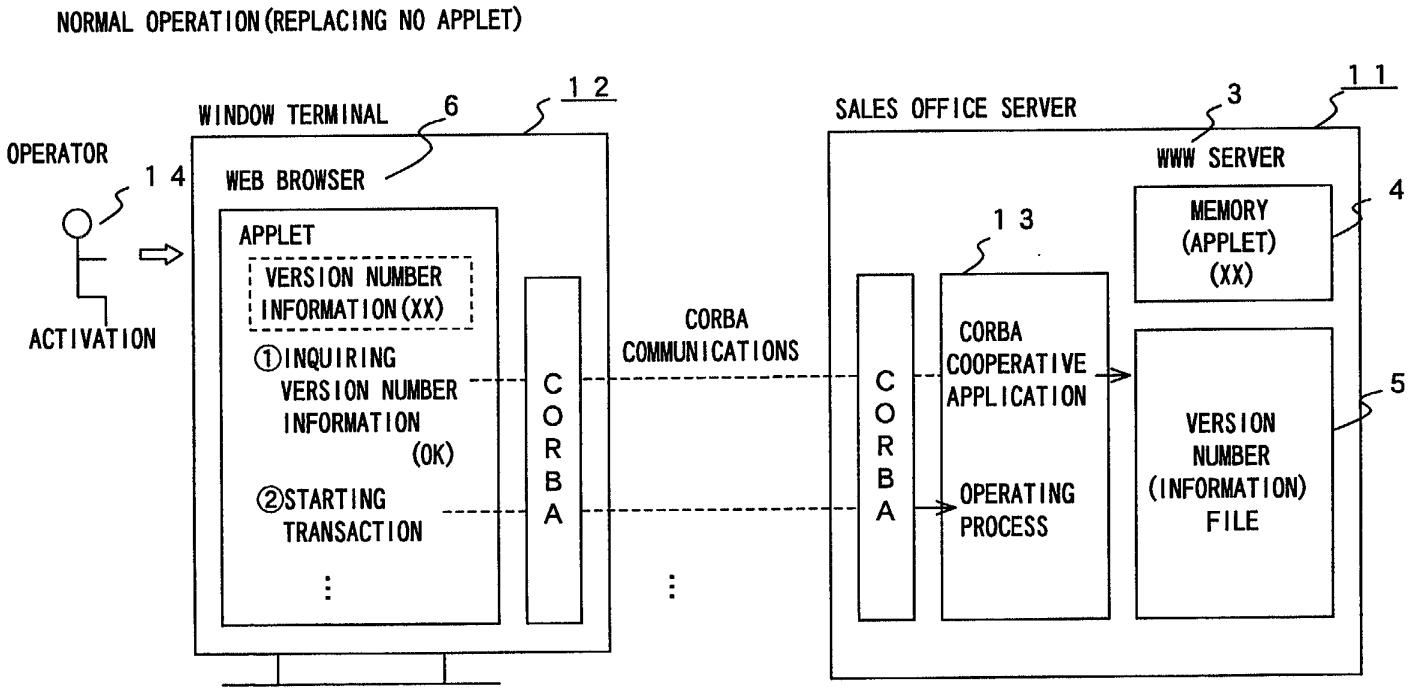

NORMAL OPERATION(REPLACING NO APPLET)
WINDOW TERMINAL
6
12
OPERATOR
14
WEB BROWSER
APPLET
VERSION NUMBER INFORMATION(XX)
①INQUIRING VERSION NUMBER INFORMATION (OK)
②STARTING TRANSACTION
CORBA
ACTIVATION
CORBA COMMUNICATIONS
SALES OFFICE SERVER
11
3
WWW SERVER
13
MEMORY (APPLET) (XX)
4
CORBA
CORBA COOPERATIVE APPLICATION
OPERATING PROCESS
VERSION NUMBER (INFORMATION) FILE
5

FIG. 9

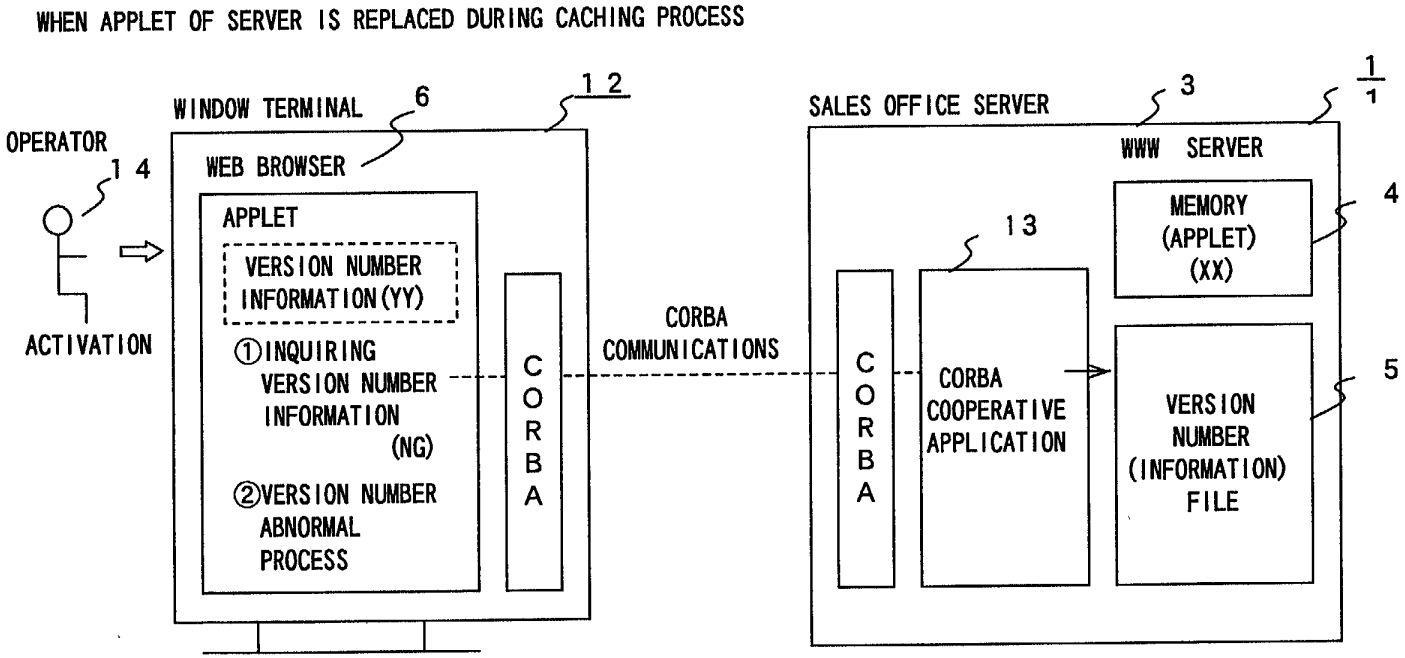

WHEN APPLET OF SERVER IS REPLACED DURING CACHING PROCESS
WINDOW TERMINAL
6
12
OPERATOR
14
WEB BROWSER
APPLET
VERSION NUMBER INFORMATION(YY)
①INQUIRING VERSION NUMBER INFORMATION (NG)
②VERSION NUMBER ABNORMAL PROCESS
ACTIVATION
CORBA
CORBA COMMUNICATIONS
SALES OFFICE SERVER
3
1
1
WWW SERVER
13
CORBA
CORBA COOPERATIVE APPLICATION
MEMORY (APPLET) (XX)
4
VERSION NUMBER (INFORMATION) FILE
5

FIG.10

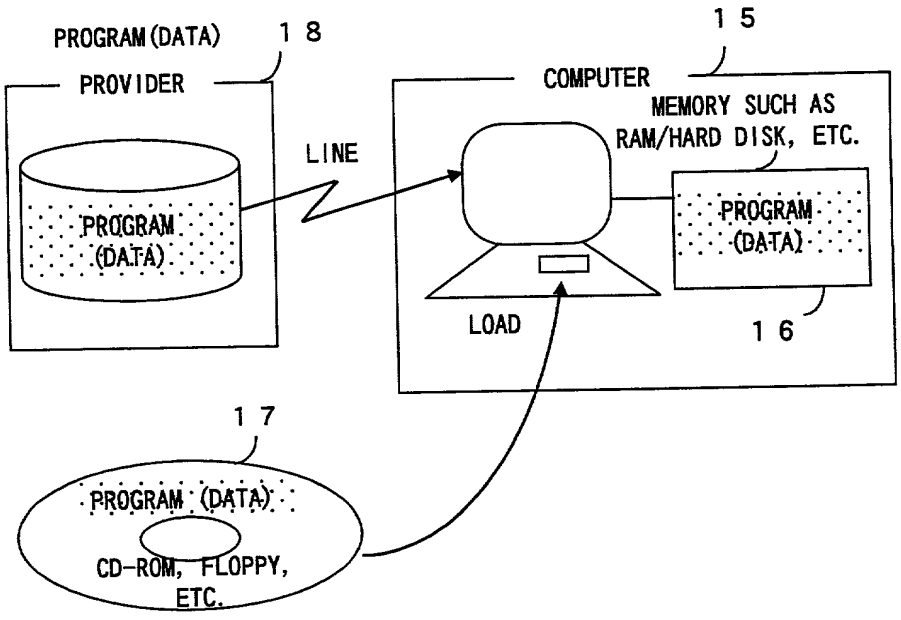

PROGRAM(DATA)
18
PROVIDER
PROGRAM
(DATA)
LINE
15
COMPUTER
MEMORY SUCH AS
RAM/HARD DISK, ETC.
PROGRAM
(DATA)
16
LOAD
17
PROGRAM (DATA)
CD-ROM, FLOPPY,
ETC.

FIG. 11

PTO/SB/106 (8-96)
Approved for use through 9/30/98. OMB 0651-0032
Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

Declaration and Power of Attorney For Patent Application

特許出願宣言書及び委任状

Japanese Language Declaration

日本語宣言書

下記の氏名の発明者として、私は以下の通り宣言します。

As a below named inventor, I hereby declare that:

私の住所、私書箱、国籍は下記の私の氏名の後に記載された通りです。

My residence, post office address and citizenship are as stated next to my name.

下記の名称の発明に関して請求範囲に記載され、特許出願している発明内容について、私が最初かつ唯一の発明者（下記の氏名が一つの場合）もしくは最初かつ共同発明者であると（下記の名称が複数の場合）信じています。

I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled

SYSTEM AND METHOD FOR AUTOMATICALLY SETTING APPLET LATEST VERSION

上記発明の明細書（下記の欄でx印がついていない場合は、本書に添付）は、

the specification of which is attached hereto unless the following box is checked:

☐ __月__日に提出され、米国出願番号または特許協定条約国際出願番号を_____とし、
（該当する場合）_____ に訂正されました。

☐ was filed on _____
as United States Application Number or
PCT International Application Number
_____ and was amended on
_____ (if applicable).

私は、特許請求範囲を含む上記訂正後の明細書を検討し、内容を理解していることをここに表明します。

I hereby state that I have reviewed and understand the contents of the above identified specification, including the claims, as amended by any amendment referred to above.

私は、連邦規則法典第３７編第１条５６項に定義されるとおり、特許資格の有無について重要な情報を開示する義務があることを認めます。

I acknowledge the duty to disclose information which is material to patentability as defined in Title 37, Code of Federal Regulations, Section 1.56.

PTO/SB/106 (8-96)
Approved for use through 9/30/98. OMB 0651-0032
Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

Japanese Language Declaration
(日本語宣言書)

私は、米国法典第３５編１１９条(a)-(d)項又は３６５条(b)項に基き下記の、米国以外の国の少なくとも一ヵ国を指定している特許協力条約３６５(a)項に基ずく国際出願、又は外国での特許出願もしくは発明者証の出願についての外国優先権をここに主張するとともに、優先権を主張している、本出願の前に出願された特許または発明者証の外国出願を以下に、枠内をマークすることで、示しています。

I hereby claim foreign priority under Title 35, United States Code, Section 119 (a)-(d) or 365(b) of any foreign application(s) for patent or inventor's certificate, or 365(a) of any PCT International application which designated at least one country other than the United States, listed below and have also identified below, by checking the box, any foreign application for patent or inventor's certificate, or PCT International application having a filing date before that of the application on which priority is claimed.

Prior Foreign Application(s)
外国での先行出願

(Number) (番号)	(Country) (国名)	(Day/Month/Year Filed) (出願年月日)	Priority Not Claimed 優先権主張なし
2000-054056	JAPAN	29th/February/2000	☐
			☐

私は、第３５編米国法典１１９条(e)項に基いて下記の米国特許出願規定に記載された権利をここに主張いたします。

I hereby claim the benefit under Title 35, United States Code, Section 119(e) of any United States provisional application(s) listed below.

(Application No.) (出願番号)	(Filing Date) (出願日)	(Application No.) (出願番号)	(Filing Date) (出願日)

私は、下記の米国法典第３５編１２０条に基いて下記の米国特許出願に記載された権利、又は米国を指定している特許協力条約３６５条(c)に基ずく権利をここに主張します。また、本出願の各請求範囲の内容が米国法典第３５編１１２条第１項又は特許協力条約で規定された方法で先行する米国特許出願に開示されていない限り、その先行米国出願書提出日以降で本出願書の日本国内または特許協力条約国際提出日までの期間中に入手された、連邦規則法典第３７編１条５６項で定義された特許資格の有無に関する重要な情報について開示義務があることを認識しています。

I hereby claim the benefit under Title 35, United States Code, Section 120 of any United States application(s), or 365(c) of any PCT International application designating the United States, listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States or PCT International application in the manner provided by the first paragraph of Title 35, United States Code Section 112, I acknowledge the duty to disclose information which is material to patentability as defined in Title 37, Code of Federal Regulations, Section 1.56 which became available between the filing date of the prior application and the national or PCT International filing date of application.

(Application No.) (出願番号)	(Filing Date) (出願日)	(Status: Patented, Pending, Abandoned) (現況：特許許可済、係属中、放棄済)

私は、私自身の知識に基ずいて本宣言書中で私が行なう表明が真実であり、かつ私の入手した情報と私の信じるところに基ずく表明が全て真実であると信じていること、さらに故意になされた虚偽の表明及びそれと同等の行為は米国法典第１８編第１００１条に基ずき、罰金または拘禁、もしくはその両方により処罰されること、そしてそのような故意による虚偽の声明を行なえば、出願した、又は既に許可された特許の有効性が失われることを認識し、よってここに下記のごとく宣誓を致します。

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

PTO/SB/106 (8-96)
Approved for use through 9/30/96. OMB 0651-0032
Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

Japanese Language Declaration
(日本語宣言書)

委任状： 私は下記の発明者として、本出願に関する一切の手続きを米特許商標局に対して遂行する弁理士または代理人として、下記の者を指名いたします。（弁護士、または代理人の氏名及び登録番号を明記のこと）

POWER OF ATTORNEY: As a named inventor, I hereby appoint the following attorney(s) and/or agent(s) to prosecute this application and transact all business in the Patent and Trademark Office connected therewith *(list name and registration number)*

James D. Halsey, Jr., 22,729; Harry John Staas, 22,010; David M. Pitcher, 25,908; John C. Garvey, 28,607; J. Randall Beckers, 30,358; William F. Herbert, 31,024; Richard A. Gollhofer, 31,106; Mark J. Henry, 36,162; Gene M. Garner II, 34,172; Michael D. Stein, 37,240; Paul I. Kravetz, 35,230; Gerald P. Joyce, III, 37,648; Todd E. Marlette, 35,269; Harlan B. Williams, Jr., 34,756; George N. Stevens, 36,938; Michael C. Soldner, 41,455; Norman L. Ourada, 41,235; Kevin R. Spivak, P-43,148; and William M. Schertler, 35,348 (agent)

書類送付先

Send Correspondence to:

STAAS & HALSEY
700 Eleventh Street, N.W.
Suite 500
Washington, D.C. 20001

直接電話連絡先： （名前及び電話番号）

Direct Telephone Calls to: *(name and telephone number)*

STAAS & HALSEY
(202) 434-1500

唯一または第一発明者名	Full name of sole or first inventor Koujirou OGURA
発明者の署名 日付	Inventor's signature Date Koujirou Ogura November 1, 2000
住所	Residence Kawasaki, Japan
国籍	Citizenship Japan
私書箱	Post Office Address c/o FUJITSU LIMITED, 1-1, Kamikodanaka 4-chome, Nakahara-ku, Kawasaki-shi, Kanagawa 211-8588, Japan
第二共同発明者	Full name of second joint inventor, if any
第二共同発明者 日付	Second inventor's signature Date
住所	Residence
国籍	Citizenship
私書箱	Post Office Address

（第三以降の共同発明者についても同様に記載し、署名をすること）

(Supply similar information and signature for third and subsequent joint inventors.)